Design Space Exploration on Heterogeneous SoC: The H.264 encoder case-study

Mokhtar Bouain*, Rabie Ben Atitallah*, Ahmed Ben Atitallah†, Nouri Masmoudi† and Jean-luc Dekeyser‡
*LAMIH, INRIA Lille Nord Europe, Université de Valenciennes et du Hainaut-Cambrésis
†LETI, École Nationale d'Ingnieurs de Sfax
‡LIFL, INRIA Lille Nord Europe, Université de Lille1

***Abstract*—This paper deals with design challenges intrinsic in nowadays heterogeneous System on Chip (SoC) embedding multicore processors and reconfigurable fabrics. These new architectures yield to software/hardware execution model supporting dynamicity and parallelism management at the same level which comes with additional complexity in the design flow. This new hardware paradigm opens many opportunities for research since there is a lack of CAD tools to map efficiently applications on such technology taking into consideration different trade-offs between performance, power consumption, Quality of service (QoS), etc. In this paper, we present preliminary results taking as example the H.264/AVC encoder using the Zynq 7000 based platform.**

I. Introduction

Due to the growing computation rates of nowadays high performance embedded applications, using heterogeneous architectures becomes an incontrovertible solution to meet performance, flexibility, and quality of service (QoS) goals. In such systems, multi-core processors provide high computation rates while the reconfigurable logic offers high performance per watt and adaptability to the application constraints. Designers could exploit the existing partitioning in the application (i.e. hardware-software and parallel-sequential hardware) which leads to several feasible implementations whose performances vary with the chosen partitioning. With the management of the parallelism intrinsic in the application, reconfigurable technology could offer better performances comparing to CPUs or GPUs up to 10x [1] at lower frequencies. Furthermore, the design can be customized at runtime using Dynamic Partial Reconfiguration (DPR) feature offered by recent FPGA technologies. Using heterogeneous systems allows to adapt the architecture according to the application constraints and thus to optimize hardware resources. All these benefits emphasize hardware designers to redirect their efforts on heterogeneous embedded computing.

Recently, heterogeneous system on chip are becoming available on the market. The Xilinx Zynq 7000 Extensible Processing Platform is an example of such circuit embedding a dual Cortex A9 processor and tens of thousands of programmable gate arrays. The Zynq 7000 [1] combines the software programmability of a Processor with the hardware programmability of an FPGA, resulting in unrivaled levels of system performance, flexibility, scalability while providing system benefits in terms of power reduction, lower cost with fast time to market. Undoubtedly, the essential features of systems to combine software and hardware programmability (sequential and parallel) and to reconfigure themselves (at the hardware or the software level) at run-time comes with additional complexity in the different design flow steps. In this work, we present a case-study through the H.264 encoder implemented on the Zynq 7000 based platform in order to show the broadness of the architectural solution space and the real requirement of tools to help designers to take the appropriate implementation choices.

[1]http://www.xilinx.com/

II. The target system

Next generation system on chip will support heterogeneous, dynamic, and parallel execution model allowed by recent embedded architectures. These architectures gather multicore processors and reconfigurable fabrics supporting Dynamic Partial Reconfiguration (DPR) features. The system designer will face several implementation choices such as sequential software, parallel software, hardware/software, parallel hardware, and dynamic hardware. Fig. 1 illustrates our target system based on the Xilinx Zynq 7000 platform. The software can be carried out using the dual core Cortex A9 with an operating frequency up to 700 MHz while the hardware can be implemented on static or dynamic reconfigurable regions. Today, several design metrics should be considered while exploring the different architectural alternatives in order to meet the global system requirements. An efficient and fast design space exploration (DSE) of such systems needs a set of tools capable of estimating performance at the hardware and software levels.

III. Experimental Results

Our case-study consists in implementing the H.264/AVC encoder using the Zynq 7000 based platform. The main functional blocks of the application are shown in Fig. 2. The algorithm implements a 16x16/4x4 intra prediction of the H.264/AVC encoder allowing to take the decision about the best prediction mode presenting the lowest Sum of Absolute Difference (SAD). A complete transformation Integer Cosine Transform (ICT) is then performed for the frequency domain transition. After that coefficients are quantized and encoded using Context Adaptive variable Length code Codes (CAVLC). The inverse chain transform allows the reconstruction of the

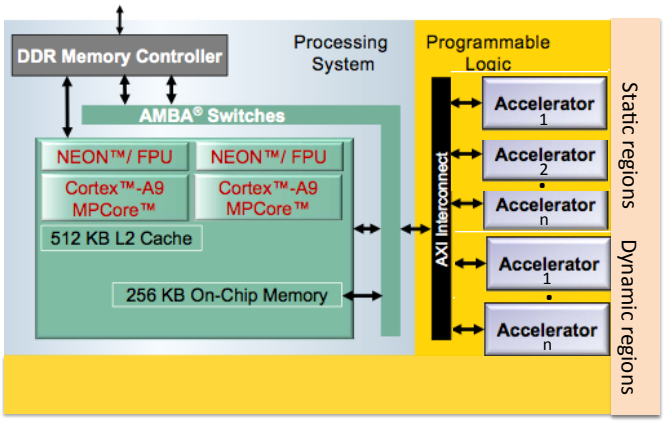

Fig. 1. Heterogeneous SoC system

whole frame. When the reconstruction is completed, the frame is filtered using the Deblocking filter [2].

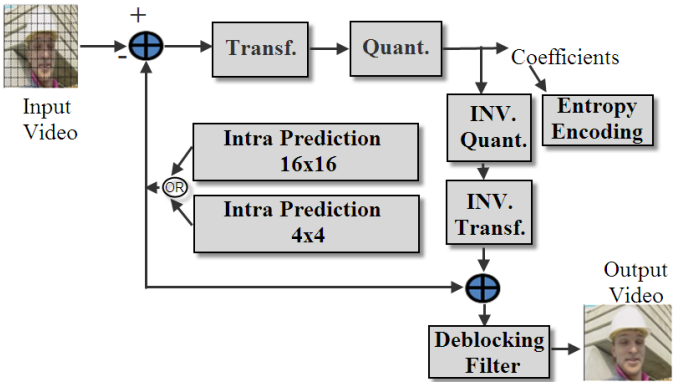

Fig. 2. 16x16/4x4 Intra prediction algorithm of the H.264/AVC encoder

As a first experiment, we carry out a sequential software implementation using a Cotex A9 processor with a frequency of 667 MHz. The used video test sequence is the "Foreman CIF" (352x288 pixels) coded in a 4:2:0 YUV format and containing 300 frames. In the second experiment, we used the Pthread library in order to implement a parallel version for the H.264/AVC encoder. The algorithm of parallelization consists in assigning the processing of each frame to a thread, it means that when the first thread processes the frame i, the second thread processes the frame $i+1$. The mapping of threads on the available two processors is managed by the scheduler of Linux operating system. For the third experiment, we select to implement the ICT task as a hardware accelerator allowing a software/hardware implementation.

Fig. 3 represents the profiling of H.264 encoder with the sequential implementation. Based on this result, the highest processing time corresponds respectively to the intra prediction and encoding of luminance and the intra prediction and encoding of chrominance with a percentage of 64% and 16%. The software profiling can help designer to detect the most time consuming tasks and to select the appropriate candidate for a hardware implementation.

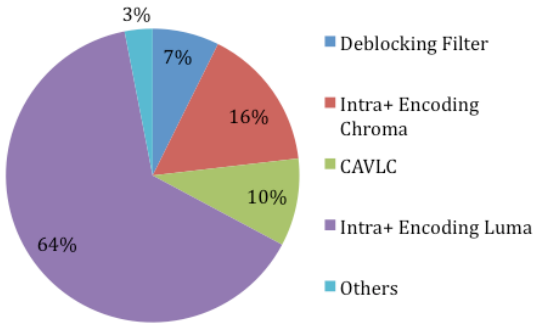

Fig. 3. Software profiling 16x16/4x4 of the H.264/AVC encoder

Table I reports the experimental results for the different implementations in terms of execution time per frame in ms, the encoder performance in $frame/s$ (f/s), the power consumption in mW, an the total energy in j. Several comments can be drawn. First, the parallel software implementation offers the better performance of 23 f/s. It improves the execution time with 47% comparing to the sequential implementation at the price of power increase of 27%. The hardware/software implementation presents the lowest power consumption. In general adding hardware accelerators improves the system performances, however this is not the case of the ICT hardware. The main reason is the communication latency between the software and the hardware parts. In fact, we used a simple connection between the Cortex A9 and the ICT accelerator. As future work, we propose to explore more efficient buses available on the Zync 7000 platform, the AXI4 for high performance memory mapped interfaces and the AXI4-Stream for high speed streaming data.

TABLE I
H.264/AVC ENCODER IMPLEMENTATIONS

Implementation	Execution time (ms)	Number of f/s	Power (mW)	Energy (J)
Sequential software	79	12	470	0.44
Parallel software	42	23	600	0.57
Software/Hardware	119	8	440	0.418

IV. Conclusion

This paper presents preliminary results while exploring the H.264/AVC encoder implementations onto a heterogeneous SoC. Our future works will cover the exploration of efficient communication buses to improve the performances and extending the execution model to support parallel hardware and dynamic behavior.

References

[1] S. Asano, T. Maruyama, and Y. Yamaguchi. Performance Comparison of FPGA, GPU AND CPU in Image Processing. In *19th IEEE International Conference on Field Programmable Logic and Applications, FPL*, Prague, Czech Republic, Aug. 2009.

[2] M. Bouain, A. B. Atitallah, and N. Masmoudi. SW Implementation of video coding Using Open Virtual Platform. In *The 10 th International Multi-Conference on Systems, Signals Devices*, Hammamet, Tunisia, Mar. 2013.